

RECEIVED
CENTRAL FAX CENTER

09/588,725
DOCKET NO. F-10190

5

DEC 18 2006

REMARKS

Claims 1-16 are all the claims presently pending in the application. Claims 1, 10, and 14 are amended to more clearly define the invention. Claims 1, 10, and 14 are independent.

These amendments are made only to more particularly point out the invention for the Examiner and not for narrowing the scope of the claims or for any reason related to a statutory requirement for patentability.

Applicant also notes that, notwithstanding any claim amendments herein or later during prosecution, Applicant's intent is to encompass equivalents of all claim elements.

Entry of this §1.116 Amendment is proper. Since the Amendments above narrow the issues for appeal and since such features and their distinctions over the prior art of record were discussed earlier, such amendments do not raise a new issue requiring a further search and/or consideration by the Examiner. As such, entry of this Amendment is believed proper and Applicant earnestly solicits entry. No new matter has been added.

Claims 1-16 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the Iizuka et al. reference.

This rejection is respectfully traversed in the following discussion.

I. THE CLAIMED INVENTION

An exemplary embodiment of the present invention, as defined by, for example, independent claim 1 is directed to a task system that includes a storage for storing an event identifier for each event of a plurality of events, a task control device for creating a task based on at least one of the events, and a task processing device for executing a plurality of tasks. Whereupon completing a first task of the plurality of tasks, the task processing device initiates a search for another event identifier, and if the another event identifier is the same as an event identifier corresponding to the first task, then processes a second task, corresponding to the another event identifier using the same resource used by the first task.

Conventional task processing systems have problems improving the speed of the task processing because it has been impossible to schedule the tasks beforehand and it has been necessary to generate overhead for acquiring and releasing the resources that are necessary for task processing.

Further, conventional task processing systems are not suitable for processing a large number of small tasks, because a lot of overhead for task switching has been necessary.

09/588,725
DOCKET NO. F-10190

6

In stark contrast, an exemplary embodiment provides a task system that stores event identifiers and that, upon completion of a first task, searches those event identifiers to process a second task that has an event identifier that is the same as the first task and which uses the same resource that was used by the first task. In this manner, overhead for acquiring and releasing resources is reduced which increases the speed of task processing. (Page 3, lines 3-8). For example, it becomes possible to process a large number of small tasks at a higher speed than has conventionally been possible. (Page 3, lines 13-15).

This feature of an exemplary embodiment of the invention consists in that after completing a first task, a task processing device searches whether or not an event identifier of an event corresponding to the completed task is registered in a storage, and if it is registered, the task processing device processes a second task using a resource used by the first task, thereby reducing the acquiring and releasing of resources that are necessary for processing the same task.

In other words, an exemplary embodiment of the present invention identifies an event using an event identifier. In this manner, after a first task is completed, a search for a second event identifier which is the same as the first identifier, and, therefore, performs the same task, may be found and the resource used by the first task may be immediately used by the second task.

II. THE 35 U.S.C. § 112 REJECTIONS

The Office Action rejects claims 1-16 under 35 U.S.C. § 112, first and second paragraph. While Applicant submits that such would be clear to one of ordinary skill in the art to allow them to know the metes and bounds of the invention, taking the present Application as a whole, to speed prosecution claims 1, 10, and 14 have been amended in accordance with Examiner To's very helpful suggestions.

In view of the foregoing, the Examiner is respectfully requested to withdraw these rejections.

III. THE PRIOR ART REJECTION

The Examiner alleges that it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the teachings of the Iizuka et al. reference to form the claimed invention. Applicant submits, however, that it would not have been

09/588,725
DOCKET NO. F-10190

7

obvious to modify the teachings of the Iizuka et al. reference and that there are elements of the claimed invention which are neither taught nor suggested by the Iizuka et al. reference.

The Examiner has failed to present a *prima facie* case for obviousness.

"To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine the teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations." M.P.E.P. Section 2143.

In the present case, the Examiner has failed to both: 1) provide any suggestion or motivation that is found either in the reference or in the knowledge generally available to make modify the disclosure to form the claimed invention; and 2) provide a prior art reference that teaches or suggests all of the claim limitations.

The Iizuka et al. reference does not teach or suggest: 1) a storage for storing an event identifier for each event of a plurality of events (claims 1 and 10); 2) a task controller for creating a task based on at least one of the events (claim 1); 3) whereupon completing a first task of the plurality of tasks, the task processing device initiates a search for another event identifier, and if the another event identifier is the same as an event identifier corresponding to the first task, then processes a second task, corresponding to the another event identifier using the same resource used by the first task (claims 1 and 10); 4) determining whether a first event identifier corresponding to a first task is the same as a second event identifier corresponding to a second task (claim 14); 5) deleting the first event identifier corresponding to the first task from an event storing unit upon completion of the processing of the first task (claim 14); and 6) processing a second task with the first task resource, if the second event identifier is the same as the first event identifier (claim 14). As explained above, these features are important for reducing the acquiring and releasing of resources that are necessary for processing tasks.

Rather, and in stark contrast, the Iizuka et al. reference discloses a resource access controller that controls access to a resource based upon ID data that corresponds to tasks and sub-tasks that are related data processing. For example, when a task requests access to a resource, the Iizuka et al. reference discloses comparing the ID data from the task to the ID data that was previously stored in a file control block 15, and if, the ID data corresponds, then

09/588,725
DOCKET NO. F-10190

8

access to the resource is granted to that task. In this manner, tasks that perform related data processing, as identified by the ID data (i.e. main tasks and corresponding sub-tasks) are allowed access to the same resource, but tasks which do not perform relate data processing, as identified by the ID data (i.e. main tasks and unrelated main and/or sub-tasks) are denied access to the resource.

The Iizuka et al. reference only discloses storing ID data which indicates tasks which are main tasks and sub-tasks (i.e. related data processing).

A brief primer on the distinction between an event and a task is believed to be helpful in view of the allegations made in the Office Action.

An event is "An action or occurrence, often generated by the user, to which a program might respond – for example, key presses, button clicks, or mouse movements." (Computer Dictionary, Third Edition, Microsoft Press, 1977, page 181).

In stark contrast, a task is "A stand-alone application or a sub-program that is run as an independent entity." (Id, page 461).

Events and tasks are generally related in that tasks may be programs that are event-driven. Event-driven programs, such as, for example, the Windows operating system, is a "type of programming in which the program constantly evaluates and responds to sets of events, such as key presses or mouse movements." (Id. Page 182). In other words, in event-driven programs, tasks are performed in response to events.

In particular, it is clear that an "event" is not the same as a "task."

In the Response to Arguments section of the October 17, 2006, Office Action, the Examiner appears to allege that an "event" and a "task" are the same. "In addition, Iizukad disclosed that one or more tasks can access to the same resource only if their IDs (event identifiers) are identical." (page 9, paragraph 31). Clearly, the Examiner is basing the rejection upon the allegation that the processing IDs which are disclosed by the Iizuka et al. reference correspond to the claimed event identifiers. As will be explained in detail below, the processing IDs that are disclosed by the Iizuka et al. reference are not event identifiers.

Further, the Examiner's allegation in this regard stretches the claim language beyond any reasonable interpretation and certainly well beyond the plain meaning of the claim language. The allegation by the Examiner that the processing IDs that are disclosed by the Iizuka et al. reference correspond to the claimed event identifiers fails to provide the claim language with their plain meaning.

09/588,725
DOCKET NO. F-10190

9

“[D]uring examination the USPTO must give claims their broadest reasonable interpretation. This means that the words of the claim must be given their plain meaning unless applicant has provided a clear definition in the specification. . . . Ordinary, simple English words whose meaning is clear and unquestionable, absent any indication that their use in a particular context changes their meaning, are construed to mean exactly what they say.” (M.P.E.P. Section 2111.01.I.)

“In the absence of an express intent to impart a novel meaning to the claim terms, the words are presumed to take on the ordinary and customary meanings attributed to them by those of ordinary skill in the art.” (M.P.E.P. Section 2111.01.II)

As explained below, the Examiner has failed to give the claims their broadest reasonable interpretation by alleging that the processing IDs that are disclosed by the Iizuka et al. reference correspond to the claimed event identifiers.

The Examiner alleges that the Iizuka et al. reference discloses a storage for storing an event identifier for each event of a plurality of events and attempts to support the allegation by referring to item 15 of Figure 1, col. 3, lines 31-37, and col. 4, lines 51-59.

However, contrary to the Examiner’s allegations the Iizuka et al. reference does not teach or suggest anything at all about events, let alone event identifiers as recited by the independent claims.

Item 15 of Figure 1 of the Iizuka et al. reference discloses “a table for recording the control data used when the controller 14 makes access to the file unit 13. As shown in Figure 2, the items to be recorded in the FCB 15 are a resource name of a resource to be accessed, a processing ID which is essential to the present invention, and a waiting queue for recording a task waiting in the access queue.” (col. 3, lines 31 – 37).

Each processing ID is generated by the processing ID generator 12 in response to a request by a main task 10. Each processing ID includes unique identifying data that corresponds to related data processings. The related data processings being tasks that are related to each other in that the form part of the same data processing. For example, sub-tasks perform data processing that is related to the data processing of the corresponding main task. In other words, the processing IDs that are stored in the file control block 15 identify tasks that are related to each other.

Contrary to the Examiner’s allegations, the file control block 15 does not teach or suggest anything at all about events, let alone event identifiers as recited by the independent

09/588,725
DOCKET NO. F-10190

10

claims.

Clearly, item 15 of Figure 1 and col. 3, lines 31-37 do not support the Examiner's allegation.

Similarly, contrary to the Examiner's allegation, col. 4, lines 51-59 do not support the Examiner's allegation.

Rather, col. 4, lines 51-59 disclose that the controller 14 executes processing for releasing the resource for the task. The controller 14 checks whether or not a task is recorded in the waiting queue in the file control block 15. "In other words, the controller 14 checks whether or not there is a task queueing up for the assignment of a given resource to a task." (col. 4, lines 55 - 57). Thus, col. 4, lines 51-59 only describes tasks, not events, let alone event identifiers.

Clearly, the Iizuka et al. reference does not teach or suggest anything at all about events, let alone event identifiers as recited by the independent claims.

The Examiner also alleges that the Iizuka et al. reference discloses a task control device for creating a task based on at least one of the plurality of events. The Examiner refers to figure 1, col. 3, lines 12-18, and col. 3, lines 47-58, in an attempt to support the Examiner's allegations.

However, contrary to the Examiner's allegations, the Iizuka et al. reference does not teach or suggest anything at all about creating any task at all, let alone creating a task based upon at least one of a plurality of events.

Col. 3, lines 12-18 describes a processing ID generator 12 that generates processing ID data. It does not teach or suggest anything at all about creating any task at all, let alone creating a task based upon at least one of a plurality of events.

Col. 3, lines 47-58 describe the processing flow of the main task 10 and the sub task 11 as illustrated by the flowcharts of Figures 8A and 8B. It describes executing a data processing operation (executing main task 10 and sub-task 11), requesting the processing ID generator 12 to generate a processing ID, how that processing ID is generated (with reference to Figure 5), and that the processing ID is stored. Col. 3, lines 47-48 does not teach or suggest anything at all about creating any task at all, let alone creating a task based upon at least one of a plurality of events.

The Examiner also alleges that the Iizuka et al. reference discloses initiating a search for another event identifier, and if the event identifier is the same as the event identifier of the

09/588,725
DOCKET NO. F-10190

11

first task, then processing a second task, corresponding to the another event identifier, using a resource used by the first task. The Examiner attempts to support this allegation by referring to Figures 8A and 8B, and col. 5, lines 28-35).

However, Figures 8A and 8B and col. 5, lines 28-35, do not support the Examiner's allegations.

Rather, as explained above, Figures 8A and 8B describes executing a data processing operation (executing main task 10 and sub-task 11), requesting the processing ID generator 12 to generate a processing ID, how that processing ID is generated (with reference to Figure 5), and that the processing ID is stored. (Col. 3, lines 47-54). Clearly, Figures 8A and 8B do not teach or suggest anything at all regarding an event, let alone an event identifier, searching for an event identifier, or, if the event identifier is the same as event identifier of the first task, then processing a second task, corresponding to the another event identifier, using a resource used by the first task, as claimed.

Col. 5, lines 28-35 also does not support the Examiner's allegation.

Rather, col. 5, lines 28-35 explains that, if the processing ID assigned to the main task and recorded in the file control block 15 has locked a resource, that a sub-task 11 can also access this same resource, providing that the same processing ID is assigned to the sub-task 11. "In other words, two or more tasks, which co-operate to perform data processing, can access the same resource only if their IDs are identical."

Col. 5, lines 28-35 does not teach or suggest anything at all regarding an event, let alone an event identifier, searching for an event identifier, or if the event identifier is the same as event identifier of the first task, then processing a second task, corresponding to the another event identifier, using a resource used by the first task, as claimed.

Therefore, the Iizuka et al. reference does not teach or suggest each and every element of the claimed invention and the Examiner is respectfully requested to withdraw this rejection.

IV. FORMAL MATTERS AND CONCLUSION

In view of the foregoing amendments and remarks, Applicant respectfully submits that claims 1-16, all the claims presently pending in the Application, are patentably distinct over the prior art of record and are in condition for allowance. The Examiner is respectfully requested to pass the above application to issue at the earliest possible time.


09/588,725
DOCKET NO. F-10190

12

Should the Examiner find the Application to be other than in condition for allowance, the Examiner is requested to contact the undersigned at the local telephone number listed below to discuss any other changes deemed necessary in a telephonic or personal interview.

The Commissioner is hereby authorized to charge any deficiency in fees or to credit any overpayment in fees to Attorney's Deposit Account No. 50-0481.

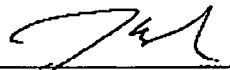
Respectfully Submitted,

Date: 12/18/06
James E. Howard, Esq.
Registration No. 39,715

McGinn Intellectual Property Law Group, PLLC
8321 Old Courthouse Rd., Suite 200
Vienna, Virginia 22182
(703) 761-4100
Customer No. 21254

CERTIFICATION OF FACSIMILE TRANSMISSION

I hereby certify that I am filing this Amendment After-Final Rejection Under 37 CFR §1.116 by facsimile with the United States Patent and Trademark Office to Examiner Jennifer N. To, Group Art Unit 2195 at fax number (571) 273-8300 this 18th day of December, 2006.


James E. Howard, Esq.
Registration No. 39,715